

**LAPORAN HASIL PRAKTIKUM  
PEMROGRAMAN WEB & MOBILE I**



**NAMA : HUSSAIN SYACH NURALAM**  
**NIM : 193020503029**  
**KELAS : A**  
**MODUL : PHP MYSQL DATABASE**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS PALANGKA RAYA  
2021**

# **BAB I**

## **PENDAHULUAN**

### **1.1 Tujuan**

Tujuan dari dilaksanakannya praktikum Pemrograman Web dan Mobile, modul III tentang PHP MySQL Database adalah sebagai berikut.

- 1) Mahasiswa mampu membuat program yang bisa menyimpan data dalam jumlah yang banyak.
- 2) Mahasiswa mampu membuat program yang bisa mengolah data yang tersimpan dalam database.

### **1.2 Landasan Teori**

PHP (PHP: Hypertext Preprocessor) adalah sebuah bahasa pemrograman server side scripting yang bersifat open source. Sebagai sebuah scripting language, PHP menjalankan instruksi pemrograman saat proses runtime. Hasil dari instruksi tentu akan berbeda tergantung data yang diproses. PHP merupakan bahasa pemrograman server-side, sehingga script dari PHP nanti akan diproses di dalam server. Jenis server yang sering digunakan bersama dengan PHP ialah seperti Apache, Nginx, dan LiteSpeed. Selain itu, PHP juga merupakan bahasa pemrograman open source. Pengguna dapat dengan bebas memodifikasi dan mengembangkan sesuai dengan kebutuhannya (Awwaabiin, 2020).

HTML adalah singkatan dari HyperText Markup Language yaitu bahasa pemrograman standar yang digunakan untuk membuat sebuah halaman web, yang kemudian dapat diakses untuk menampilkan berbagai informasi di dalam sebuah penjelajah web Internet (Browser). HTML dapat juga digunakan sebagai link link antara file-file dalam situs atau dalam komputer dengan menggunakan localhost, atau link yang menghubungkan antar situs dalam dunia internet.

Dokumen HTML adalah file yang diakhiri dengan ekstensi .html atau .htm. Ekstensi file ini bisa dilihat dengan menggunakan web browser apa pun (seperti Google Chrome, Safari, atau Mozilla Firefox). Browser tersebut membaca file

HTML dan me-render kontennya sehingga user internet bisa melihat dan membacanya. Biasanya, rata-rata situs web menyertakan sejumlah halaman HTML yang berbeda-beda. Contohnya, beranda utama, halaman ‘tentang kami’, halaman kontak yang semuanya memiliki dokumen HTML terpisah. Masing-masing halaman HTML terdiri atas seperangkat tags (bisa disebut juga elements), yang mengacu pada building block halaman website. Tag tersebut membuat hirarki yang menyusun konten hingga menjadi bagian, paragraf, heading, dan block konten lainnya (Christy, 2020).

Data dalam database MySQL disimpan dalam tabel. Tabel adalah kumpulan data terkait yang berisi kolom dan baris. Database sangat berguna untuk menyimpan informasi dalam kategori. Sebuah contoh akan diberikan dalam modul praktik ini, tabel berisi data tentang karyawan, produk, pelanggan, dan pesanan

#### **1.2.1) Membuka Koneksi**

Sebelum mengakses data di database MySQL, harus dilakukan koneksi ke server database MySQL. Berikut ini adalah contoh kode program untuk terhubung ke server mySQL

```
<?php

$servername = "localhost";
$username = "username";
$password = "password";

// Membuat hubungan
$conn = new mysqli($servername, $username, $password);

// Memeriksa hubungan
if ($conn->connect_error){
```

Gambar 1.1 Query Buka Koneksi

```
die("Connection failed: " . $conn->connect_error);

}
echo "Connected successfully";
?>
```

Gambar 1.2 Query Buka Koneksi

Apabila kode diatas tidak berhasil, kemungkinan variabel `$connect_error` sudah tidak tersedia pada versi PHP yang kita gunakan, maka gunakan kode program berikut:

```
<?php

$servername = "localhost";
$username = "username";
$password = "password";

// Membuat Hubungan
$conn = mysqli_connect($servername, $username, $password);

// Memeriksa
Hubungan if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
echo "Connected successfully";
?>
```

Gambar 1.3 Query Buka Koneksi

Setelah menggunakan data dalam database, sebaiknya harus menutup koneksi ke server. Untuk menutupnya digunakan kode program sebagai berikut:

```
mysqli_close($conn);
```

Gambar 1.4 Query Menutup Koneksi

### 1.2.2) Membuat Database

Database MySQL juga dapat dibuat dengan menggunakan kode program PHP. Program ini berisi pernyataan SQL "BUAT DATABASE". Berikut ini adalah contoh program yang digunakan untuk membuat database "myDB":

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";

// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password);
// Memeriksa Koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}

// Membuat Database
$sql = "CREATE DATABASE myDB";
if (mysqli_query($conn, $sql)) {
echo "Database created successfully";
} else {
echo "Error creating database: " . mysqli_error($conn);
}
mysqli_close($conn);
?>
```

Gambar 1.5 Query Membuat Database

### 1.2.3) Membuat Tabel

Pembuatan tabel pada bahasa pemrograman PHP juga menggunakan statement SQL, yaitu statement "CREATE TABLE". Contoh yang akan

diberikan adalah pembuatan tabel MyGuests. Statement pembuatan tabel MyGuests adalah sebagai berikut:

```
CREATE TABLE MyGuests (  
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY, firstname  
VARCHAR(30) NOT NULL, lastname VARCHAR(30) NOT NULL,  
email VARCHAR(50), reg_date TIMESTAMP  
)
```

Gambar 1.6 Query Create Table

Berikut adalah contoh program untuk membuat tabel diatas:

```
<?php  
$servername = "localhost";  
$username = "username";  
$password = "password";  
$dbname = "myDB";  
  
// Membuat Koneksi  
$conn = mysqli_connect($servername, $username, $password,  
$dbname);  
// Memeriksa koneksi if (!$conn){  
die("Connection failed: " . mysqli_connect_error());  
}  
  
// SQL untuk membuat table  
$sql = "CREATE TABLE MyGuests (  
)";
```

Gambar 1.7 Query Create Table

```

id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL, lastname VARCHAR(30)
NOT NULL,
email VARCHAR(50),
reg_date TIMESTAMP
)”,
if (mysqli_query($conn, $sql)) {
echo "Table MyGuests created successfully";
} else {
echo "Error creating table: " . mysqli_error($conn);
}
mysqli_close($conn);
?>

```

Gambar 1.8 Query Create Table

#### 1.2.4) Memasukan Data ke Database

Untuk memasukkan data kedalam tabel di database, terdapat beberapa aturan sintaks yang harus diikuti:

- 1) Query SQL harus diberikan kutip dalam PHP.
- 2) Nilai string didalam query SQL harus diberikan kutip.
- 3) Nilai numeris tidak harus diberikan kutip.
- 4) Nilai NULL tidak harus diberikan kutip.

Pernyataan INSERT INTO digunakan untuk menambahkan baris ke tabel MySQL. Kueri untuk menambahkan data adalah sebagai berikut: MASUKKAN KE nama\_tabel (kolom 1, kolom 2, kolom 3 ...) Adapun contoh kode program PHP untuk penambahan data tersebut adalah sebagai berikut:

```

INSERT INTO table_name (column1, column2, column3,...)
VALUES (value1, value2, value3,...)

```

Adapun contoh kode program PHP untuk menambahkan data tersebut adalah sebagai berikut:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Membuat Koneksi

$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa Koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";
if (mysqli_query($conn, $sql)) {
echo "New record created successfully";
} else {
echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}
mysqli_close($conn);
?>
```

Gambar 1.9 Query Input Data

Jika kita melakukan perintah INSERT pada tabel yang menggunakan field dengan opsi AUTO\_INCREMENT, kita bisa mendapatkan ID dari baris yang terakhir diinputkan. Caranya menggunakan fungsi `mysqli_insert_id($conn)`; nilai kembalian dari fungsi ini akan memberikan id dari record atau baris terakhir yang diinputkan (Dosen Teknik Informatika, 2021).



Untuk menambahkan record dengan jumlah yang banyak, kita bisa menggunakan fungsi `mysqli_multi_query($conn, $sql)`, berikut adalah contoh program untuk menambahkan record dengan jumlah yang banyak:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);

// Memeriksa Koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}

$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";

$sql .= "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('Mary', 'Moe', 'mary@example.com')";

$sql .= "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('Julie', 'Dooley', 'julie@example.com')";

if (mysqli_multi_query($conn, $sql)) {
echo "New records created successfully";
} else {
echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}

mysqli_close($conn);
?>
```

Gambar 1.10 Query Input Data

### 1.2.5) Mengambil Data dari Database

Untuk mengambil data, statement SQL yang digunakan adalah `SELECT nama_kolom() FROM nama_tabel`, atau kita bisa menggunakan karakter `*` untuk memilih semua kolom yang ada pada tabel. Pengambilan data dengan kriteria tertentu bisa dilakukan dengan menggunakan statement `WHERE` setelah `nama_tabel`. Adapun contoh kode program untuk mengambil data dari database adalah sebagai berikut:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB"; // Membuat koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa Koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
$sql = "SELECT id, firstname, lastname FROM MyGuests"; $result =
mysqli_query($conn, $sql);
if (mysqli_num_rows($result) > 0) {
// Menampilkan data pada setiap baris while($row =
mysqli_fetch_assoc($result)) {
echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " .
$row["lastname"]. "<br>";
}
} else {
echo "0 results";
}
mysqli_close($conn);
?>
```

Gambar 1.11 Query Mengambil Data

### 1.2.6) Menghapus Data dari Database

Statement DELETE digunakan untuk menghapus baris data atau records dari tabel. Clause WHERE digunakan untuk menspesifikasikan baris yang akan dihapus. Jika statement DELETE digunakan tanpa menggunakan clausa WHERE, maka semua record yang ada pada tabel akan dihapus.

id	namadepan	namabelakang	email	reg_date
1	John	Doe	john@example.com	2014-10-22 14:26:15
2	Mary	Moe	mary@example.com	2014-10-23 10:22:30
3	Julie	Dooley	julie@example.com	2014-10-26 10:48:23

Gambar 1.12 Data Pada Database MyGuest

Jika ingin menghapus data dengan nama depan Julie, maka kode program untuk menghapus data tersebut adalah sebagai berikut:

```
<?php
$servername = "localhost";
$username = "username"
$password = "password";
$dbname = "myDB";
// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
// sql untuk menghapus record
$sql = "DELETE FROM MyGuests WHERE id=3";
if (mysqli_query($conn, $sql)) {
echo "Record deleted successfully";
} else {
echo "Error deleting record: " . mysqli_error($conn);
}
```

Gambar 1.13 Query Hapus Data

```
mysqli_close($conn);  
?>
```

Gambar 1.14 Query Hapus Data

### 1.2.7) Update Data Di Dalam Database

Untuk melakukan perubahan data di dalam database, statement UPDATE digunakan, yaitu sebagai berikut:

```
UPDATE nama_tabel  
SET kolom1=nilai1, kolom2=nilai2, ...  
WHERE kolom_penentu=nilai_penentu ...
```

Jika statement ini digunakan untuk update data ke dua pada tabel “MyGuest” diatas untuk mengganti nama belakang dari “Moe” menjadi “Doe”, maka kode program untuk update tersebut adalah sebagai berikut:

```
<?php  
$servername = "localhost";  
$username = "username";  
$password = "password";  
$dbname = "myDB";  
  
// Membuat Koneksi  
$conn = mysqli_connect($servername, $username, $password,  
$dbname);  
  
// Memeriksa Koneksi if (!$conn){  
die("Connection failed: " . mysqli_connect_error());  
}  
$sql = "UPDATE MyGuests SET lastname='Doe' WHERE id=2";  
if (mysqli_query($conn, $sql)) {  
echo "Record updated successfully";  
} else {
```

Gambar 1.15 Query Update Data

```
echo "Error updating record: " . mysqli_error($conn);  
}  
mysqli_close($conn);  
?>
```

Gambar 1.16 Query Update Data

## BAB II

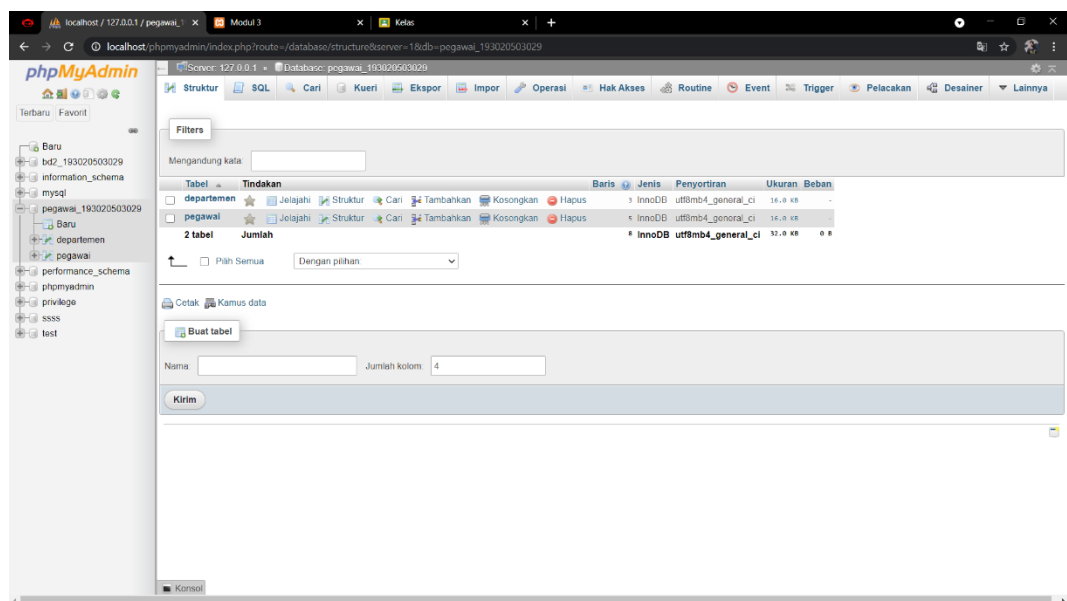
### PEMBAHASAN

#### 2.1 Langkah Kerja

- 1) Buatlah database pada MySql untuk menyimpan data-data pegawai beserta relasi tabelnya.
- 2) Buatlah program untuk menyimpan, menghapus dan mengubah data-data pegawai tersebut.

#### 2.2 Pembahasan

Untuk pembuatan database dapat dilakukan dengan menggunakan mariadb via command prompt, php myadmin, maupun sqlyog. Pada laporan ini, pembuatan database dilakukan dengan menggunakan php myadmin. Berikut adalah gambar dan query dari database yang dibuat di php myadmin.



Gambar 2.1 Database pegawai\_193020503029

```
-- phpMyAdmin SQL Dump
-- version 5.1.0
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Waktu pembuatan: 12 Apr 2021 pada 06.15
-- Versi server: 10.4.18-MariaDB
-- Versi PHP: 8.0.3

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `pegawai_193020503029`
--

--
-- Struktur dari tabel `departemen`
--

CREATE TABLE `departemen` (
  `id_departemen` int(3) UNSIGNED ZEROFILL NOT NULL,
  `nama_departemen` varchar(30) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Dumping data untuk tabel `departemen`
--

INSERT INTO `departemen` (`id_departemen`, `nama_departemen`) VALUES
```

Gambar 2.2 Query Database pegawai\_193020503029

```
(001, 'Sales Marketing'),
(002, 'HRD'),
(003, 'Information & Technology');

--
-- Struktur dari tabel `pegawai`
--

CREATE TABLE `pegawai` (
  `id` int(4) UNSIGNED ZEROFILL NOT NULL,
  `fname` varchar(50) DEFAULT NULL,
  `lname` varchar(30) DEFAULT NULL,
  `tgl_lahir` date DEFAULT NULL,
  `alamat` varchar(50) DEFAULT NULL,
  `jk` varchar(1) DEFAULT NULL,
  `gaji` int(10) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Dumping data untuk tabel `pegawai`
--

INSERT INTO `pegawai` (`id`, `fname`, `lname`, `tgl_lahir`, `alamat`, `jk`, `gaji`) VALUES
(0001, 'Husin', 'Muralam', '2001-01-01', 'Jl. Raden Patah', 'L', 2500000),
(0002, 'Yunita', 'Amelia', '2000-02-03', 'Jl. Bukit Indah', 'P', 2375000),
(0003, 'Aditya', 'Ramadhani', '1999-09-09', 'Jl. RTA Milono', 'L', 2750000),
(0004, 'Zefanya', 'Kristina', '2001-03-05', 'Jl. Rajawali', 'P', 2600000),
(0005, 'Firdaus', 'Zailani', '2000-12-11', 'Jl. Yos Sudarso', 'L', 2500000);

--
-- Indexes for dumped tables
--

--
-- Indeks untuk tabel `departemen`
--
ALTER TABLE `departemen`
```

Gambar 2.3 Query Database pegawai\_193020503029

```

-- ADD PRIMARY KEY (`id_departemen`);
--
-- Indeks untuk tabel `pegawai`
--
ALTER TABLE `pegawai`
| ADD PRIMARY KEY (`id`);
--
-- AUTO_INCREMENT untuk tabel yang dibuang
--
--
-- AUTO_INCREMENT untuk tabel `departemen`
--
ALTER TABLE `departemen`
| MODIFY `id_departemen` int(3) UNSIGNED ZEROFILL NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;
--
-- AUTO_INCREMENT untuk tabel `pegawai`
--
ALTER TABLE `pegawai`
| MODIFY `id` int(4) UNSIGNED ZEROFILL NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=8;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

Gambar 2.4 Query Database pegawai\_193020503029

Langkah praktikum selanjutnya ialah membuat untuk menyimpan, menghapus dan mengubah data-data pegawai tersebut. Yang pertama adalah menyimpan data-data pegawai. Sebelum membuat program php functionnya, maka harus dibuat terlebih dulu program yang dapat mengoneksikan antara function-function php dengan HTML. Berikut adalah sintaks dari program koneksi PHP.

```

D: > xampp > htdocs > 193020503029_modul3 > koneksi.php
1  <?php
2
3      $host="localhost";
4      $user="root";
5      $password="";
6      $db="pegawai_193020503029";
7
8      $kon = mysqli_connect($host,$user,$password);
9      //if ($kon){
10         // echo "Database MYSQL <b>berhasil</b> dikoneksikan<br>";
11     //}else {
12         // echo "Database MYSQL <b>gagal</b> dikoneksikan<br>";
13     //}
14
15     $hasil=mysqli_select_db($kon,$db);
16     //if ($hasil){
17         // echo "Database $db berhasil dipilih";
18     //}else {
19         //echo "Database $db gagal dipilih";
20     //}
21

```

Gambar 2.5 Sintaks Koneksi Database pegawai\_193020503029



Dalam penggunaan PHP di atas, terdapat sintaks-sintaks yang perlu ditaati. Berikut adalah sintaks-sintaksnya.

- 1) `<?php` : Sintaks ini adalah kode wajib untuk membuka program PHP.
- 2) `Echo` : Sintaks ini adalah sebuah perintah untuk menampilkan teks.
- 3) `?>` : Sintaks ini adalah kode untuk mengakhiri PHP dan wajib digunakan saat digabung dengan bahasa pemrograman lain seperti HTML (Awwaabiin, 2020).
- 4) `$host` adalah Hostname/IP address yang digunakan untuk mengakses MySQL.
- 5) `$user` adalah user yang ada di database MySQL.
- 6) `$password` adalah password dari user yang ada di database MySQL.
- 7) `$db` adalah penunjuk yang mengarahkan PHP untuk membaca database yang akan ditampilkan.
- 8) `$kon` berfungsi sebagai penghubung antara php koneksi dengan php php lainnya yang memiliki array yang sama dengan yang ada di dalam parameter `$kon`.
- 9) `$hasil` berfungsi untuk menampilkan hasil yang nanti ditambahkan ke database melalui function-function php yang telah dibuat.
- 10) Fungsi `mysqli_connect()` merupakan fungsi yang digunakan untuk membuka koneksi ke server MySQL dan memilih database yang akan digunakan.
- 11) Fungsi `mysqli_select_db` adalah fungsi php untuk menjalankan argumen agar terkoneksi ke database mysql. Isi argumen tersebut adalah sebuah value nama database yang telah dibuat.

Setelah itu, pertama-tama buatlah index yang berfungsi untuk memasukan data HTML yang nantinya akan dijalankan pada web melalui localhost. Index adalah file yang menjadi indeks situs web. Ketika kita membuka website dengan memanggil nama domain dari website tersebut, maka file akan terbuka secara otomatis. Indeks HTML adalah file default dan file terpenting dari situs web. File index harus ditempatkan di home atau direktori root situs web. File `index.html` juga harus ditempatkan di setiap direktori dan subdirektori dari

direktori situs web. Hal ini dilakukan untuk alasan keamanan guna melindungi website dari peretas yang tidak bertanggung jawab.

Berikut adalah gambar dari sintaks-sintaks pada index.

```
<?php
require("koneksi.php");
$data = mysqli_query($kon, "SELECT * FROM pegawai");
?>

<!DOCTYPE html>
<html>
<head>
|   <title>Modul 3</title>
</head>
<body>
|   <h3>Daftar Data Pegawai</h3>
|   <a href="penyimpanan.php">Tambah Data</a>

|   <table border="1px">
|   <tr>
|       <th>ID Pegawai</th>
|       <th>Nama Depan</th>
|       <th>Nama Belakang</th>
|       <th>Tanggal Lahir</th>
|       <th>Alamat</th>
|       <th>Jenis Kelamin</th>
|       <th>Gaji</th>
|       <th>Aksi</th>
|   </tr>

|   <?php while($row = mysqli_fetch_assoc($data)){ ?>
|   <tr>
|       <td><?php echo $row['id'] ?></td>
|       <td><?php echo $row['fname'] ?></td>
|       <td><?php echo $row['lname'] ?></td>
|       <td><?php echo $row['tgl_lahir'] ?></td>
|       <td><?php echo $row['alamat'] ?></td>
|       <td><?php echo $row['jk'] ?></td>
|       <td><?php echo $row['gaji'] ?></td>
|       <td>
|           <a href="update.php?id=<?php echo $row['id']; ?>">Edit</a> |
|           <a href="delete.php?id=<?php echo $row['id']; ?>">Hapus</a>
|       </td>
|   </tr>
```

Gambar 2.6 Sintaks Index Database pegawai\_193020503029

```
</tr>
<?php
} // end while

mysqli_close($kon); // menutup koneksi ke database
?>
</table>
</body>
</html>
```

Gambar 2.7 Sintaks Index Database pegawai\_193020503029

Dari kedua gambar di atas, didapat beberapa sintaks dasar yang harus dimasukan di dalam program PHP agar program tersebut dapat dieksekusi. Berikut adalah penjelasannya.

- 1) `<!DOCTYPE html>`; adalah suatu deklarasi yang digunakan untuk mengidentifikasi jenis dokumen HTML yang digunakan sehingga browser dapat menentukan bagaimana memperlakukan kode tersebut.
- 2) `<html>`; adalah tag utama yang digunakan untuk memulai dokumen HTML.
- 3) `<head>`; adalah tag utama yang digunakan untuk membuat bagian kepala dokumen.
- 4) `<title>`; adalah tag utama yang digunakan untuk mengatur judul web.
- 5) `<body>`; adalah tag utama yang digunakan untuk membuat bagian tubuh dokumen (Muhardian, 2020).
- 6) Atribut `border` digunakan untuk mengatur ketebalan dari garis tepi (`border`) dari tabel. Jika atribut ini tidak ditulis, maka web browser akan menampilkan tabel tanpa garis tepi. Nilai dari atribut ini berupa angka yang diukur dalam satuan pixel. Jika anda memberikan nilai `border="1"`, maka web browser akan menampilkan garis tepi sebesar 1 pixel pada sisi atas, bawah, kiri dan kanan tabel (Andre, 2013).
- 7) Elemen `<tr>` digunakan untuk mendefinisikan pembuatan baris pada tabel
- 8) Elemen `<td>` digunakan untuk membuat kolom atau sel di setiap baris pada tabel
- 9) Tag `<a>` digunakan untuk keperluan hypertext, yakni text yang ketika diklik akan pindah ke halaman lainnya (Fadullah, 2018).
- 10) Atribut `href` biasanya digunakan berdampingan dengan tag `<a>` fungsinya adalah untuk membuat sebuah link. `href` sendiri berfungsi untuk mengarahkan atau mengalihkan link ke alamat tujuan yang diletakkan di bagian `value`.
- 11) `$row` adalah variabel yang akan menampung hasil fungsi `mysql_fetch_row()`. Hasil dari fungsi `mysql_fetch_row()` berupa tipe data array dengan key merujuk kepada urutan kolom, dan `value`-nya (nilai dari array) adalah isi dari kolom.

- 12) Fungsi `mysqli_fetch_assoc()` adalah untuk mengembalikan nilai berupa associative array. Selain mengembalikan nilai, fungsi `mysqli_fetch_assoc()` secara otomatis menaikkan posisi pointer dari variabel `$query` pada tiap pemanggilan.
- 13) Fungsi `mysql_close` pada PHP adalah untuk menghentikan koneksi php ke server mysql dengan cara otomatis menggunakan fungsi `mysql_close()` setelah instruksi atau query mysql selesai dilakukan.

Berikutnya adalah membuat functionnya. Yang pertama adalah pembuatan function penyimpanan yang digunakan untuk insert data dan menyimpan data yang telah diinput. Berikut adalah sintaks-sintaksnya.

```
#!/php
include("koneksi.php");

if(isset($_POST['submit'])){
    $fname = $_POST['fname'];
    $lname = $_POST['lname'];
    $tgl_lahir = $_POST['tgl_lahir'];
    $alamat = $_POST['alamat'];
    $jk = $_POST['jk'];
    $gaji = $_POST['gaji'];

    // query insert data ke database dalam tabel anggota
    $query = "INSERT INTO pegawai SET fname='$fname', lname= '$lname', tgl_lahir='$tgl_lahir', alamat='$alamat', jk='$jk', gaji='$gaji'";
    if(mysqli_query($kon, $query)){
        header("Location: index.php");
    }else{
        echo "Tambah data gagal";
    }
}

mysqli_close($kon); // menutup koneksi ke database
```

Gambar 2.8 Sintaks Function Penyimpanan

```
<!DOCTYPE html>
<html>
<head>
    <title>Input Data Pegawai</title>
</head>
<body>
    <h3>Tambah Data Pegawai</h3>
    <form action="penyimpanan.php" method="post">
        <table>
            <tr><td>Nama Depan</td><td><input type="text" name="fname"></td></tr>
            <tr><td>Nama Belakang</td><td><input type="text" name="lname"></td></tr>
            <tr><td>Tanggal Lahir</td><td><input type="date" name="tgl_lahir"></td></tr>
            <tr><td>Alamat</td><td><textarea name="alamat" rows="3" cols="20"></textarea></td></tr>
            <tr><td>Jenis Kelamin</td><td><input type="text" name="jk"></td></tr>
            <tr><td>Gaji</td><td><input type="number" name="gaji"></td></tr>
        </table>
        <br>
        <input type="submit" name="submit" value="Tambah Data">
    </form>
</body>
</html>
```

Gambar 2.9 Sintaks Function Penyimpanan

- 1) Fungsi PHP `include()` merupakan fungsi yang digunakan untuk menyertakan file php lain ke dalam suatu program PHP. Hal sangat membantu proses pemrograman karena tidak perlu menulis program PHP secara berulang-ulang, cukup dalam satu file saja (Indra, 2014).
- 2) Fungsi `isset` pada PHP biasanya digunakan untuk mengecek form dari HTML. Sedangkan variabel `$_POST` digunakan jika pengiriman data dari form menggunakan method POST. Tujuan dari fungsi `isset` pada php ini untuk menghindari error `undefined variable` (variabel tidak didefinisi) (Lukman, 2019).
- 3) Form adalah penunjuk bahwa data berbentuk seperti formulir saat akan diisi. Form action merupakan form handling yang nantinya data yang dapat dihandling melalui action. Action method ini berupa `$_POST` (Hadi, 2015).
- 4) Elemen `<tr>` digunakan untuk mendefinisikan pembuatan baris pada table.
- 5) Elemen `<td>` digunakan untuk membuat kolom atau sel di setiap baris pada table.
- 6) Elemen `<br>` digunakan untuk memberikan break pada teks sehingga teks yang ditampilkan akan memiliki jarak dengan teks di atas maupun di bawahnya.
- 7) Type digunakan untuk penunjuk tipe data yang digunakan saat akan menginput data ke dalam database.

Berikutnya ada function delete yang ditunjukkan seperti gambar berikut.

```
<?php
require_once("koneksi.php");

$id = $_GET['id'];
$query = "DELETE FROM pegawai WHERE id = $id"; // query hapus data

if(mysqli_query($kon, $query)){
    header("Location: index.php"); // redirect ke index.php
}else{
    echo "Hapus data gagal";
}
```

Gambar 2.10 Sintaks Function Delete

Function ini digunakan untuk menghapus data yang ingin dihilangkan oleh pengguna program. Berikut adalah sintaks-sintaksnya.

- 1) Fungsi PHP `require()` merupakan fungsi yang digunakan untuk menyertakan file php lain ke dalam suatu program PHP. Hal sangat membantu proses pemrograman karena tidak perlu menulis program PHP secara berulang-ulang, cukup dalam satu file saja. Pada `require()`, jika file yang disertakan tidak ditemukan (karena lokasi yang salah atau memang file tidak ada) maka program atau perintah PHP selanjutnya (setelah `include`) tidak akan dijalankan (Indra, 2014).
- 2) `$id` digunakan dengan handling `$_GET`. Penggunaannya ialah sebagai penampung data-data yang merupakan bagian dari id yang ada di dalam database.
- 3) `$query` digunakan untuk menampung perintah delete data dari database berdasarkan dengan kondisi yang telah ditunjukkan yakni berdasarkan dari id.

Yang terakhir adalah update data. Hal ini digunakan agar pengguna program bisa mengedit kesalahan yang mungkin ada dalam data dari pegawai. Berikut adalah sintaks-sintaksnya.

```
<?php
include("koneksi.php");

if(isset($_POST['submit'])){
    $fname = $_POST["fname"];
    $lname = $_POST["lname"];
    $tgl_lahir = $_POST["tgl_lahir"];
    $alamat = $_POST["alamat"];
    $jk = $_POST["jk"];
    $gaji = $_POST["gaji"];
}
?>

<!DOCTYPE html>
<html>
<head>
    <title>Update Data Pegawai</title>
</head>
<body>
    <h3>Update Data</h3>
    <form action="penyimpanan.php" method="post">
        <table>
            <tr><td>Nama Depan</td><td><input type="text" name="fname"></td></tr>
            <tr><td>Nama Belakang</td><td><input type="text" name="lname"></td></tr>
            <tr><td>Tanggal Lahir</td><td><input type="date" name="tgl_lahir"></td></tr>
            <tr><td>Alamat</td><td><textarea name="alamat" rows="3" cols="20"></textarea></td></tr>
            <tr><td>Jenis Kelamin</td><td><input type="text" name="jk"></td></tr>
            <tr><td>Gaji</td><td><input type="number" name="gaji"></td></tr>
            <tr><td><input type="hidden" name="id" value="<?php echo $row['id']; ?>"></td></tr>
            <tr><td><input type="submit" name="submit" value="Edit Data"></td></tr>
        </table>
    </form>
<?php
$kon = 'SELECT * FROM pegawai';
$query = mysqli_query($kon,$db);
while($kon = mysqli_fetch_assoc($data))
?>
```

Gambar 2.11 Sintaks Function Update

```

    <?php
    | // end while
    |
    mysqli_close($kon); // menutup koneksi ke database
    ?>
</body>
</html>

```

Gambar 2.12 Sintaks Function Update

- 1) Fungsi PHP include() merupakan fungsi yang digunakan untuk menyertakan file php lain ke dalam suatu program PHP. Hal sangat membantu proses pemrograman karena tidak perlu menulis program PHP secara berulang-ulang, cukup dalam satu file saja.
- 2) Fungsi isset pada PHP biasanya digunakan untuk mengecek form dari HTML. Sedangkan variabel \$\_POST digunakan jika pengiriman data dari form menggunakan method POST. Tujuan dari fungsi isset pada php ini untuk menghindari error undefined variable (variabel tidak didefinisi).
- 3) \$\_POST digunakan untuk menampilkan data-data yang merupakan bagian yang ada di dalam database.
- 4) Elemen <tr> digunakan untuk mendefinisikan pembuatan baris pada table.
- 5) Elemen <td> digunakan untuk membuat kolom atau sel di setiap baris pada table.
- 6) Form adalah penunjuk bahwa data berbentuk seperti formulir saat akan diisi. Form action merupakan form handling yang nantinya data yang dapat dihandling melalui action. Action method ini berupa \$\_POST.
- 7) Type digunakan untuk penunjuk tipe data yang digunakan saat akan menginput data ke dalam database.
- 8) Objek form textarea digunakan untuk membuat text inputan yang bisa menampung lebih dari 1 baris inputan. Tag textarea mirip dengan tag input type text, namun memiliki kelebihan untuk menampung beberapa baris. Biasanya textarea digunakan untuk inputan yang panjang, seperti komentar, keterangan, atau catatan.

Sehingga, saat dijalankan didapat output sebagai berikut.

localhost/193020503029\_modul3/index.php

**Daftar Data Pegawai**

[Tambah Data](#)

ID Pegawai	Nama Depan	Nama Belakang	Tanggal Lahir	Alamat	Jenis Kelamin	Gaji	Aksi
0001	Husin	Nuralam	2001-01-01	Jl. Raden Patah	L	2500000	<a href="#">Edit</a> <a href="#">Hapus</a>
0002	Yunita	Amelia	2000-02-03	Jl. Bukit Indah	P	2375000	<a href="#">Edit</a> <a href="#">Hapus</a>
0003	Aditya	Ramadhani	1999-09-09	Jl. RTA Milono	L	2750000	<a href="#">Edit</a> <a href="#">Hapus</a>
0004	Zefanya	Kristina	2001-03-05	Jl. Rajawali	P	2600000	<a href="#">Edit</a> <a href="#">Hapus</a>
0005	Firdaus	Zailani	2000-12-11	Jl. Yos Sudarso	L	2500000	<a href="#">Edit</a> <a href="#">Hapus</a>

Gambar 2.13 Output Data Pegawai



## Daftar Data Pegawai

[Tambah Data](#)

ID Pegawai	Nama Depan	Nama Belakang
0001	Husin	Nuralam
0002	Yunita	Amelia
0003	Aditya	Ramadhani
0004	Zefanya	Kristina
0005	Firdaus	Zailani

Gambar 2.14 Output Tambah Data

localhost/193020503029\_modul3/penyimpanan.php

**Tambah Data Pegawai**

Nama Depan

Nama Belakang

Tanggal Lahir

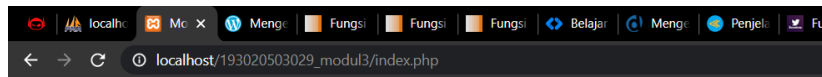
Alamat

Jenis Kelamin

Gaji

Gambar 2.15 Output Tambah Data





### Daftar Data Pegawai

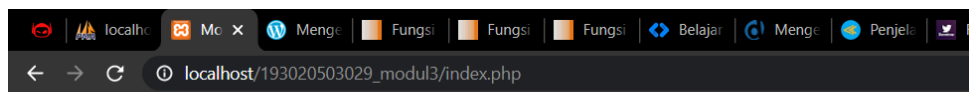
[Tambah Data](#)

ID Pegawai	Nama Depan	Nama Belakang	Tanggal Lahir	Alamat	Jenis Kelamin	Gaji	Aksi
0001	Husin	Nuralam	2001-01-01	Jl. Raden Patah	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0002	Yunita	Amelia	2000-02-03	Jl. Bukit Indah	P	2375000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0003	Aditya	Ramadhani	1999-09-09	Jl. RTA Milono	L	2750000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0004	Zefanya	Kristina	2001-03-05	Jl. Rajawali	P	2600000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0005	Firdaus	Zailani	2000-12-11	Jl. Yos Sudarso	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0008	Rosa	Rosi	2000-03-31	Jl. Diponegoro	P	2300000	<a href="#">Edit</a>   <a href="#">Hapus</a>

Gambar 2.16 Hasil Tambah Data

ID Pegawai	Nama Depan	Nama Belakang	Tanggal Lahir	Alamat	Jenis Kelamin	Gaji	Aksi
0001	Husin	Nuralam	2001-01-01	Jl. Raden Patah	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0002	Yunita	Amelia	2000-02-03	Jl. Bukit Indah	P	2375000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0003	Aditya	Ramadhani	1999-09-09	Jl. RTA Milono	L	2750000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0004	Zefanya	Kristina	2001-03-05	Jl. Rajawali	P	2600000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0005	Firdaus	Zailani	2000-12-11	Jl. Yos Sudarso	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0008	Rosa	Rosi	2000-03-31	Jl. Diponegoro	P	2300000	<a href="#">Edit</a>   <a href="#">Hapus</a>

Gambar 2.17 Hapus Data

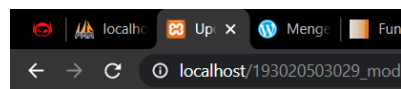


### Daftar Data Pegawai

[Tambah Data](#)

ID Pegawai	Nama Depan	Nama Belakang	Tanggal Lahir	Alamat	Jenis Kelamin	Gaji	Aksi
0001	Husin	Nuralam	2001-01-01	Jl. Raden Patah	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0002	Yunita	Amelia	2000-02-03	Jl. Bukit Indah	P	2375000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0003	Aditya	Ramadhani	1999-09-09	Jl. RTA Milono	L	2750000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0004	Zefanya	Kristina	2001-03-05	Jl. Rajawali	P	2600000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0005	Firdaus	Zailani	2000-12-11	Jl. Yos Sudarso	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>

Gambar 2.18 Hasil Hapus Data



### Update Data

Edit Data

Nama Depan

Nama Belakang

Tanggal Lahir

Alamat

Jenis Kelamin

Gaji

Gambar 2.19 Update Data

### **BAB III**

#### **KESIMPULAN**

PHP (PHP: Hypertext Preprocessor) adalah sebuah bahasa pemrograman server side scripting yang bersifat open source. Sebagai sebuah scripting language, PHP menjalankan instruksi pemrograman saat proses runtime. Hasil dari instruksi tentu akan berbeda tergantung data yang diproses. PHP merupakan bahasa pemrograman server-side, sehingga script dari PHP nanti akan diproses di dalam server. Jenis server yang sering digunakan bersama dengan PHP ialah seperti Apache, Nginx, dan LiteSpeed. Selain itu, PHP juga merupakan bahasa pemrograman open source.

HTML adalah singkatan dari HyperText Markup Language yaitu bahasa pemrograman standar yang digunakan untuk membuat sebuah halaman web, yang kemudian dapat diakses untuk menampilkan berbagai informasi di dalam sebuah penjelajah web Internet (Browser).

## DAFTAR PUSTAKA

- Andre. (2013). *Atribut border, cellpadding dan cellspacing dalam Tabel*.  
[Https://Www.Duniaikom.Com/](https://Www.Duniaikom.Com/). <https://www.duniaikom.com/belajar-html-mengenal-attribut-border-cellpadding-dan-cellspacing-dalam-tabel/>
- Awwaabiin, S. (2020). *Apa itu PHP? Pengertian, Fungsi dan Sintaks Dasarnya!*  
Niagahoster.Co.Id. <https://www.niagahoster.co.id/blog/pengertian-php/>
- Christy, A. (2020). *Apa Itu HTML? Fungsi dan Cara Kerja HTML*. Hostinger International. <https://www.hostinger.co.id/tutorial/apa-itu-html/>
- Dosen Teknik Informatika. (2021). *MODUL PRAKTIKUM PEMROGRAMAN WEB I. 1*, 60.
- Fadullah. (2018). *HTML a tag. Belajar a element*. Apacara.Com.  
<https://www.apacara.com/tutorial/html/html-a-tag.html>
- Hadi, D. A. (2015). *Penanganan Form Pada PHP*.  
<Https://Www.Malasngoding.Com/>. <https://www.malasngoding.com/belajar-php-penanganan-form-pada-php/>
- Indra, R. (2014). *Mengenal Fungsi Require Dan Include Dalam PHP*.  
<Https://Www.Dumetschool.Com/>.  
[https://www.dumetschool.com/blog/Mengenal-Fungsi-Required-Dan-Include-Dalam-PHP#:~:text=Fungsi PHP include\(\) dan require\(\) merupakan fungsi yang,cukup dalam satu file saja.](https://www.dumetschool.com/blog/Mengenal-Fungsi-Required-Dan-Include-Dalam-PHP#:~:text=Fungsi PHP include() dan require() merupakan fungsi yang,cukup dalam satu file saja.)
- Lukman. (2019). *Penjelasan Fungsi Isset Pada PHP*. <Https://Carakode.Com/>.  
<https://carakode.com/penjelasan-fungsi-isset-pada-php/>
- Muhardian, A. (2020). *Belajar HTML #02 Apa itu Tag, Elemen, dan Atribut dalam HTML*. Petanikode.Com. <https://www.petanikode.com/html-tag-elemen-attribut/>

## LAMPIRAN

```
<?php

$servername = "localhost";
$username = "username";
$password = "password";

// Membuat hubungan
$conn = new mysqli($servername, $username, $password);

// Memeriksa hubungan
if ($conn->connect_error){
```

Gambar 1.1 Query Buka Koneksi

```
die("Connection failed: " . $conn->connect_error);

}
echo "Connected successfully";
?>
```

Gambar 1.2 Query Buka Koneksi

```
<?php

$servername = "localhost";
$username = "username";
$password = "password";

// Membuat Hubungan
$conn = mysqli_connect($servername, $username, $password);

// Memeriksa
Hubungan if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
echo "Connected successfully";
?>
```

Gambar 1.3 Query Buka Koneksi

```
mysqli_close($conn);
```

Gambar 1.4 Query Menutup Koneksi

```

<?php
$servername = "localhost";
$username = "username";
$password = "password";

// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password);
// Memeriksa Koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}

// Membuat Database
$sql = "CREATE DATABASE myDB";
if (mysqli_query($conn, $sql)) {
echo "Database created successfully";
} else {
echo "Error creating database: " . mysqli_error($conn);
}
mysqli_close($conn);
?>

```

Gambar 1.5 Query Membuat Database

```

CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY, firstname
VARCHAR(30) NOT NULL, lastname VARCHAR(30) NOT NULL,
email VARCHAR(50), reg_date TIMESTAMP
)

```

Gambar 1.6 Query Create Table

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);

// Memeriksa koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}

// SQL untuk membuat table
$sql = "CREATE TABLE MyGuests (
)";
```

Gambar 1.7 Query Create Table

```
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,  
firstname VARCHAR(30) NOT NULL, lastname VARCHAR(30)  
NOT NULL,  
email VARCHAR(50),  
reg_date TIMESTAMP  
)”,  
if (mysqli_query($conn, $sql)) {  
echo "Table MyGuests created successfully";  
} else {  
echo "Error creating table: " . mysqli_error($conn);  
}  
mysqli_close($conn);  
?>
```

Gambar 1.8 Query Create Table



```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Membuat Koneksi

$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa Koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";
if (mysqli_query($conn, $sql)) {
echo "New record created successfully";
} else {
echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}
mysqli_close($conn);
?>
```

Gambar 1.9 Query Input Data

```

<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";
// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa Koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";
$sql .= "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('Mary', 'Moe', 'mary@example.com')";
$sql .= "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('Julie', 'Dooley', 'julie@example.com')";
if (mysqli_multi_query($conn, $sql)) {
echo "New records created successfully";
} else {
echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}
mysqli_close($conn);
?>

```

Gambar 1.10 Query Input Data

```

<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB"; // Membuat koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa Koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
$sql = "SELECT id, firstname, lastname FROM MyGuests"; $result =
mysqli_query($conn, $sql);
if (mysqli_num_rows($result) > 0) {
// Menampilkan data pada setiap baris while($row =
mysqli_fetch_assoc($result)) {
echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " .
$row["lastname"]. "<br>";
}
} else {
echo "0 results";
}
mysqli_close($conn);
?>

```

Gambar 1.11 Query Mengambil Data

id	namadepan	namabelakang	email	reg_date
1	John	Doe	john@example.com	2014-10-22 14:26:15
2	Mary	Moe	mary@example.com	2014-10-23 10:22:30
3	Julie	Dooley	julie@example.com	2014-10-26 10:48:23

Gambar 1.12 Data Pada Database MyGuest

```
<?php
$servername = "localhost";
$username = "username"
$password = "password";
$dbname = "myDB";
// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
// sql untuk menghapus record
$sql = "DELETE FROM MyGuests WHERE id=3";
if (mysqli_query($conn, $sql)) {
echo "Record deleted successfully";
} else {
echo "Error deleting record: " . mysqli_error($conn);
}
```

Gambar 1.13 Query Hapus Data

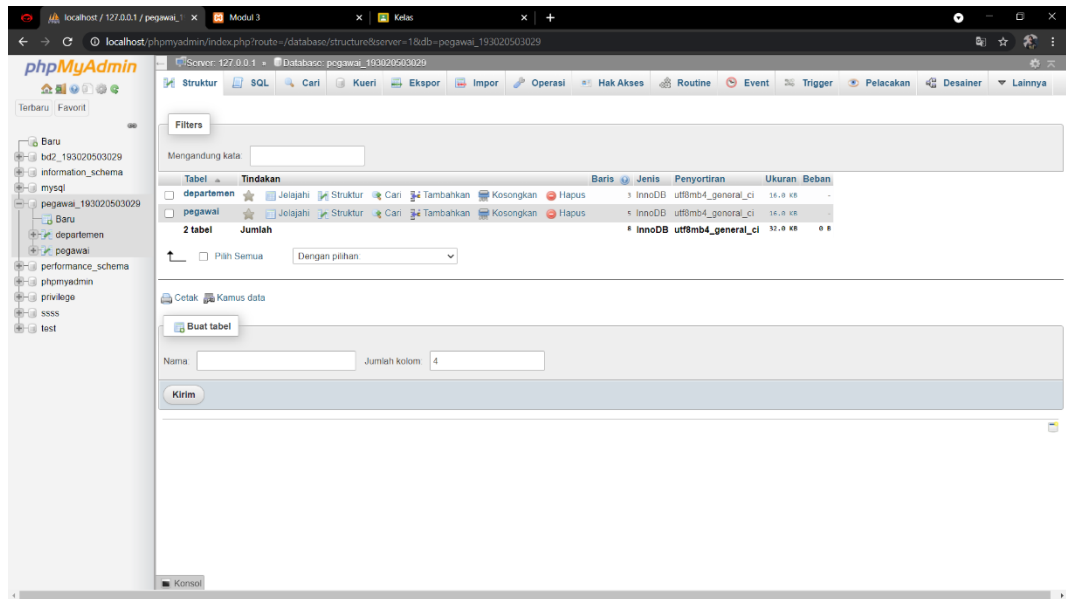
```
mysqli_close($conn);
?>
```

Gambar 1.14 Query Hapus Data

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password,
$dbname);
// Memeriksa Koneksi if (!$conn){
die("Connection failed: " . mysqli_connect_error());
}
$sql = "UPDATE MyGuests SET lastname='Doe' WHERE id=2";
if (mysqli_query($conn, $sql)) {
echo "Record updated successfully";
} else {
echo "Error updating record: " . mysqli_error($conn);
}
mysqli_close($conn);
?>
```

Gambar 1.16 Query Update Data



Gambar 2.1 Database pegawai\_193020503029

```
-- phpMyAdmin SQL Dump
-- version 5.1.0
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Waktu pembuatan: 12 Apr 2021 pada 06.15
-- Versi server: 10.4.18-MariaDB
-- Versi PHP: 8.0.3

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `pegawai_193020503029`
--

--
-- Struktur dari tabel `departemen`
--

CREATE TABLE `departemen` (
  `id_departemen` int(3) UNSIGNED ZEROFILL NOT NULL,
  `nama_departemen` varchar(30) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Dumping data untuk tabel `departemen`
--

INSERT INTO `departemen` (`id_departemen`, `nama_departemen`) VALUES
```

Gambar 2.2 Query Database pegawai\_193020503029

```

(001, 'Sales Marketing'),
(002, 'HRD'),
(003, 'Information & Technology');

-----

--
-- Struktur dari tabel `pegawai`
--
CREATE TABLE `pegawai` (
  `id` int(4) UNSIGNED ZEROFILL NOT NULL,
  `fname` varchar(50) DEFAULT NULL,
  `lname` varchar(30) DEFAULT NULL,
  `tgl_lahir` date DEFAULT NULL,
  `alamat` varchar(50) DEFAULT NULL,
  `jk` varchar(1) DEFAULT NULL,
  `gaji` int(10) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Dumping data untuk tabel `pegawai`
--
INSERT INTO `pegawai` (`id`, `fname`, `lname`, `tgl_lahir`, `alamat`, `jk`, `gaji`) VALUES
(0001, 'Husin', 'Nuralam', '2001-01-01', 'Jl. Raden Patah', 'L', 2500000),
(0002, 'Yunita', 'Amelia', '2000-02-03', 'Jl. Bukit Indah', 'P', 2375000),
(0003, 'Aditya', 'Ramadhani', '1999-09-09', 'Jl. RTA Milono', 'L', 2750000),
(0004, 'Zefanya', 'Kristina', '2001-03-05', 'Jl. Rajawali', 'P', 2600000),
(0005, 'Firdaus', 'Zailani', '2000-12-11', 'Jl. Yos Sudarso', 'L', 2500000);

--
-- Indexes for dumped tables
--

--
-- Indeks untuk tabel `departemen`
--
ALTER TABLE `departemen`

```

Gambar 2.3 Query Database pegawai\_193020503029

```

ADD PRIMARY KEY (`id_departemen`);

--
-- Indeks untuk tabel `pegawai`
--
ALTER TABLE `pegawai`
  ADD PRIMARY KEY (`id`);

--
-- AUTO_INCREMENT untuk tabel yang dibuang
--

--
-- AUTO_INCREMENT untuk tabel `departemen`
--
ALTER TABLE `departemen`
  MODIFY `id_departemen` int(3) UNSIGNED ZEROFILL NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;

--
-- AUTO_INCREMENT untuk tabel `pegawai`
--
ALTER TABLE `pegawai`
  MODIFY `id` int(4) UNSIGNED ZEROFILL NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=8;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

Gambar 2.4 Query Database pegawai\_193020503029

```

D: > xampp >htdocs > 193020503029_modul3 > koneksi.php
1  <?php
2
3      $host="localhost";
4      $user="root";
5      $password="";
6      $db="pegawai_193020503029";
7
8      $kon = mysqli_connect($host,$user,$password);
9      //if ($kon){
10         // echo "Database MYSQL <b>berhasil</b> dikoneksikan<br>";
11     //}else {
12         // echo "Database  MYSQL <b>gagal</b> dikoneksikan<br>";
13     //}
14
15     $hasil=mysqli_select_db($kon,$db);
16     //if ($hasil){
17         // echo "Database $db berhasil dipilih";
18     //}else {
19         //echo "Database $db gagal dipilih";
20     //}
21

```

Gambar 2.5 Sintaks Koneksi Database pegawai\_193020503029

```

<?php
require("koneksi.php");
$data = mysqli_query($kon, "SELECT * FROM pegawai");
?>

<!DOCTYPE html>
<html>
<head>
|   <title>Modul 3</title>
</head>
<body>
|   <h3>Daftar Data Pegawai</h3>
|   <a href="penyimpanan.php">Tambah Data</a>
|
|   <table border="1px">
|       <tr>
|           <th>ID Pegawai</th>
|           <th>Nama Depan</th>
|           <th>Nama Belakang</th>
|           <th>Tanggal Lahir</th>
|           <th>Alamat</th>
|           <th>Jenis Kelamin</th>
|           <th>Gaji</th>
|           <th>Aksi</th>
|       </tr>
|
|       <?php while($row = mysqli_fetch_assoc($data)){ ?>
|           <tr>
|               <td><?php echo $row['id'] ?></td>
|               <td><?php echo $row['fname'] ?></td>
|               <td><?php echo $row['lname'] ?></td>
|               <td><?php echo $row['tgl_lahir'] ?></td>
|               <td><?php echo $row['alamat'] ?></td>
|               <td><?php echo $row['jk'] ?></td>
|               <td><?php echo $row['gaji'] ?></td>
|               <td>
|                   <a href="update.php?id=<?php echo $row['id']; ?>">Edit</a> |
|                   <a href="delete.php?id=<?php echo $row['id']; ?>">Hapus</a>
|               </td>
|           </tr>
|       </?php >
|   </table>
|

```

Gambar 2.6 Sintaks Index Database pegawai\_193020503029



```

    </tr>
<?php
} // end while

mysqli_close($kon); // menutup koneksi ke database
?>
</table>
</body>
</html>

```

Gambar 2.7 Sintaks Index Database pegawai\_193020503029

```

<?php
include("koneksi.php");

if(isset($_POST['submit'])){
    $fname = $_POST["fname"];
    $lname = $_POST["lname"];
    $tgl_lahir = $_POST["tgl_lahir"];
    $alamat = $_POST["alamat"];
    $jk = $_POST["jk"];
    $gaji = $_POST["gaji"];

    // query insert data ke database dalam tabel anggota
    $query = "INSERT INTO pegawai SET fname='$fname', lname= '$lname', tgl_lahir='$tgl_lahir', alamat='$alamat', jk='$jk', gaji='$gaji'";
    if(mysqli_query($kon, $query)){
        header("Location: index.php");
    }else{
        echo "Tambah data gagal";
    }
}

mysqli_close($kon); // menutup koneksi ke database

```

Gambar 2.8 Sintaks Function Penyimpanan

```

<!DOCTYPE html>
<html>
<head>
    <title>Input Data Pegawai</title>
</head>
<body>
<h3>Tambah Data Pegawai</h3>
    <form action="penyimpanan.php" method="post">
        <table>
            <tr><td>Nama Depan</td><td><input type="text" name="fname"></td></tr>
            <tr><td>Nama Belakang</td><td><input type="text" name="lname"></td></tr>
            <tr><td>Tanggal Lahir</td><td><input type="date" name="tgl_lahir"></td></tr>
            <tr><td>Alamat</td><td><textarea name="alamat" rows="3" cols="20"></textarea></td></tr>
            <tr><td>Jenis Kelamin</td><td><input type="text" name="jk"></td></tr>
            <tr><td>Gaji</td><td><input type="number" name="gaji"></td></tr>
        </table>
        <br>
        <input type="submit" name="submit" value="Tambah Data">
    </form>
</body>
</html>

```

Gambar 2.9 Sintaks Function Penyimpanan

```

<?php
require_once("koneksi.php");

$id = $_GET['id'];
$query = "DELETE FROM pegawai WHERE id = $id"; // query hapus data

if(mysqli_query($kon, $query)){
    header("Location: index.php"); // redirect ke index.php
}else{
    echo "Hapus data gagal";
}
}

```

Gambar 2.10 Sintaks Function Delete

```

<?php
include("koneksi.php");

if(isset($_POST['submit'])){
    $fname = $_POST["fname"];
    $lname = $_POST["lname"];
    $tgl_lahir = $_POST["tgl_lahir"];
    $alamat = $_POST["alamat"];
    $jk = $_POST["jk"];
    $gaji = $_POST["gaji"];
}
?>

<!DOCTYPE html>
<html>
<head>
    <title>Update Data Pegawai</title>
</head>
<body>
    <h3>Update Data</h3>
    <form action="penyimpanan.php" method="post">
        <table>
            <tr><td>Nama Depan</td><td><input type="text" name="fname"></td></tr>
            <tr><td>Nama Belakang</td><td><input type="text" name="lname"></td></tr>
            <tr><td>Tanggal Lahir</td><td><input type="date" name="tgl_lahir"></td></tr>
            <tr><td>Alamat</td><td><textarea name="alamat" rows="3" cols="20"></textarea></td></tr>
            <tr><td>Jenis Kelamin</td><td><input type="text" name="jk"></td></tr>
            <tr><td>Gaji</td><td><input type="number" name="gaji"></td></tr>
            <tr><td><input type="hidden" name="id" value="<?php echo $row['id']; ?>"></td></tr>
            <tr><td><input type="submit" name="submit" value="Edit Data"></td></tr>
        </table>
    </form>
    <?php
    $kon = 'SELECT * FROM pegawai';
    $query = mysqli_query($kon,$db);
    while($kon = mysqli_fetch_assoc($data))
    ?>

```

Gambar 2.11 Sintaks Function Update

```

<?php
    // end while

    mysqli_close($kon); // menutup koneksi ke database
?>
</body>
</html>

```

Gambar 2.12 Sintaks Function Update

localhost/193020503029\_modul3/index.php

**Daftar Data Pegawai**

[Tambah Data](#)

ID Pegawai	Nama Depan	Nama Belakang	Tanggal Lahir	Alamat	Jenis Kelamin	Gaji	Aksi
0001	Husin	Nuralam	2001-01-01	Jl. Raden Patah	L	2500000	<a href="#">Edit</a> <a href="#">Hapus</a>
0002	Yunita	Amelia	2000-02-03	Jl. Bukit Indah	P	2375000	<a href="#">Edit</a> <a href="#">Hapus</a>
0003	Aditya	Ramadhani	1999-09-09	Jl. RTA Milono	L	2750000	<a href="#">Edit</a> <a href="#">Hapus</a>
0004	Zefanya	Kristina	2001-03-05	Jl. Rajawali	P	2600000	<a href="#">Edit</a> <a href="#">Hapus</a>
0005	Firdaus	Zailani	2000-12-11	Jl. Yos Sudarso	L	2500000	<a href="#">Edit</a> <a href="#">Hapus</a>

Gambar 2.13 Output Data Pegawai

localhost/193020503029

**Daftar Data Pegawai**

[Tambah Data](#)

ID Pegawai	Nama Depan	Nama Belakang
0001	Husin	Nuralam
0002	Yunita	Amelia
0003	Aditya	Ramadhani
0004	Zefanya	Kristina
0005	Firdaus	Zailani

Gambar 2.14 Output Tambah Data

localhost/193020503029\_modul3/penyimpanan.php

**Tambah Data Pegawai**

Nama Depan

Nama Belakang

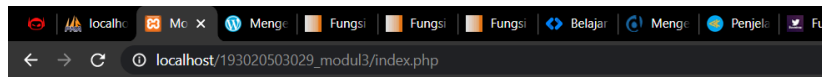
Tanggal Lahir

Alamat

Jenis Kelamin

Gaji

Gambar 2.15 Output Tambah Data



### Daftar Data Pegawai

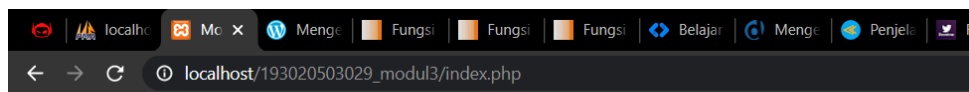
[Tambah Data](#)

ID Pegawai	Nama Depan	Nama Belakang	Tanggal Lahir	Alamat	Jenis Kelamin	Gaji	Aksi
0001	Husin	Nuralam	2001-01-01	Jl. Raden Patah	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0002	Yunita	Amelia	2000-02-03	Jl. Bukit Indah	P	2375000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0003	Aditya	Ramadhani	1999-09-09	Jl. RTA Milono	L	2750000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0004	Zefanya	Kristina	2001-03-05	Jl. Rajawali	P	2600000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0005	Firdaus	Zailani	2000-12-11	Jl. Yos Sudarso	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0008	Rosa	Rosi	2000-03-31	Jl. Diponegoro	P	2300000	<a href="#">Edit</a>   <a href="#">Hapus</a>

Gambar 2.16 Hasil Tambah Data

ID Pegawai	Nama Depan	Nama Belakang	Tanggal Lahir	Alamat	Jenis Kelamin	Gaji	Aksi
0001	Husin	Nuralam	2001-01-01	Jl. Raden Patah	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0002	Yunita	Amelia	2000-02-03	Jl. Bukit Indah	P	2375000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0003	Aditya	Ramadhani	1999-09-09	Jl. RTA Milono	L	2750000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0004	Zefanya	Kristina	2001-03-05	Jl. Rajawali	P	2600000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0005	Firdaus	Zailani	2000-12-11	Jl. Yos Sudarso	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0008	Rosa	Rosi	2000-03-31	Jl. Diponegoro	P	2300000	<a href="#">Edit</a>   <a href="#">Hapus</a>

Gambar 2.17 Hapus Data

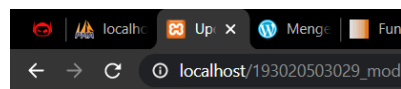


### Daftar Data Pegawai

[Tambah Data](#)

ID Pegawai	Nama Depan	Nama Belakang	Tanggal Lahir	Alamat	Jenis Kelamin	Gaji	Aksi
0001	Husin	Nuralam	2001-01-01	Jl. Raden Patah	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0002	Yunita	Amelia	2000-02-03	Jl. Bukit Indah	P	2375000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0003	Aditya	Ramadhani	1999-09-09	Jl. RTA Milono	L	2750000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0004	Zefanya	Kristina	2001-03-05	Jl. Rajawali	P	2600000	<a href="#">Edit</a>   <a href="#">Hapus</a>
0005	Firdaus	Zailani	2000-12-11	Jl. Yos Sudarso	L	2500000	<a href="#">Edit</a>   <a href="#">Hapus</a>

Gambar 2.18 Hasil Hapus Data



### Update Data

Edit Data

Nama Depan

Nama Belakang

Tanggal Lahir

Alamat

Jenis Kelamin

Gaji

Gambar 2.19 Update Data